DETAILED DESCRIPTION - An INDEPENDENT CLAIM is included for machine behavior simulation program.

USE - For simulating behavior of e.g. machine, plant, using hybrid model of the machine/plant and computer.

ADVANTAGE - Enables to model complicated mechanisms in the machine and plant, correctly and conveniently aligned with time-axis, using hybrid model.

DESCRIPTION OF DRAWING(S) - The figure shows the block diagram of the machine behavior simulator. (Drawing includes non-English language text).

hybrid simulator 101

execution unit 102

continuous group equation simulator 103

hybrid model description 104

variable value time log memory 105

CHOSEN-DRAWING: Dwg.1/9

TITLE-TERMS: HYBRID BEHAVE SIMULATE METHOD MACHINE PLANT SOLVING INTERNAL DATA EXPRESS CORRESPOND CONTINUOUS STATE TRANSITION DESCRIBE GROUP EQUATE ASSOCIATE EVENT OCCUR

DERWENT-CLASS: T01 T06

EPI-CODES: T01-J; T01-J04A; T01-J05A2F; T06-A07B;

SECONDARY-ACC-NO:
Non-CPI Secondary Accession Numbers: N2004-363233

Previous Doc Next Doc Go to Doc#

L2: Entry 28 of 28 File: DWPI Jun 24, 2004

DERWENT-ACC-NO: 2004-458027
DERWENT-WEEK: 200476

TITLE: Hybrid behavior simulation method for e.g. machine, plant, involves solving internal data expressions corresponding to continuous state transition-descriptive group equations associated with event occurrence

INVENTOR: KONDO, K

PATENT-ASSIGNEE:

ASSIGNEE	CODE
TOSHIBA KK	TOKE
KONDO K	KONDI

PRIORITY-DATA: 2002JP-0344228 (November 27, 2002)

PATENT-FAMILY:

PUB-NO	PUB-DATE	LANGUAGE	PAGES	MAIN-IPC
JP 2004178300 A	June 24, 2004		019	G06F019/00
US 20040158442 A1	August 12, 2004		000	G06F017/50
CN 1503188 A	June 9, 2004		000	G06G007/48

APPLICATION-DATA:

PUB-NO	APPL-DATE	APPL-NO	DESCRIPTOR
JP2004178300A	November 27, 2002	2002JP-0344228	
US20040158442A1	November 26, 2003	2003US-0721544	
CN 1503188A	November 27, 2003	2003CN-1118652	

INT-CL (IPC): G05 B 17/00; G06 F 17/13; G06 F 17/50; G06 F 19/00; G06 G 7/48

ABSTRACTED-PUB-NO: JP2004178300A
BASIC-ABSTRACT:

NOVELTY - A table relating conditions and objects associated with the switching of continuous group equations describing state transitions of the machine hybrid model, is produced. The table is referred according to event occurrence for determining the associated equations. The internal data expressions corresponding to the determined equations, are solved in alignment with time-axis, for expressing output behavior.

23. A method for simulating the behavior of a digital circuit, the digital circuit having a plurality of connected circuit elements, comprising the steps of

defining inputs to said circuit elements,

defining output signal levels of outputs of said circuit elements, said outputs of each circuit elements having a causal relationship with the inputs of of each said same element,

tracing the signal value of at least one of said circuit element inputs and outputs for at least a selected change of condition of a logical combination of said connected circuit element inputs and outputs,

displaying messages describing said traced circuit element inputs and outputs at times when said selected condition is attained,

selectively displaying said message whenever at least a portion of a memory element is updated with changed data,

evaluating the occurrence of at least one selected Boolean relationship among said combination of inputs and outputs,

displaying said trace message whenever the Boolean relationship is satisfied,

parsing said Boolean relationships into elemental conditions,

transforming said elemental conditions into event detection circuitry, and

inserting said event detection circuitry into said circuit, the behavior of which is to be simulated.

Previous Doc Next Doc Go to Doc#

L2: Entry 24 of 28 File: USPT Feb 16, 1988

DOCUMENT-IDENTIFIER: US 4725971 A
TITLE: Digital system simulation method and apparatus

Detailed Description Text (24):
Referring to FIG. 9, when tracing is to be implemented, the operator provides the control 10 with the necessary information regarding the tracing procedure, including the functional expression, the satisfaction of which will trigger the trace message. The processing of the tracing command is initialized at the command level of the apparatus by parsing for example the Boolean expression of the events initiating a trace, into a stack, in Reverse Polish Notation (RPN). This is indicated at 130. The "RPN stack" is then processed to transform the parsed Boolean expressions into a network of event detectors, storage update watch blocks, and Boolean evaluators. This is indicated at 132. These elements become part of the defined circuit, the behavior of which is to be simulated, and are inserted into that circuit as would be any other circuit element. This is indicated at 134. The inclusion of these additional monitoring circuit elements is, however, transparent to the user, so that he is not aware of their inclusion into the circuit.

CLAIMS:

11. A method of simulating the behavior of a digital circuit having a plurality of connected circuit elements, the method of comrpising the steps of

defining inputs to said circuit elements

defining output signal levels for outputs of said circuit elements,

establishing a causal relationship between each said circuit element inputs and outputs,

tracing the signal value of at least one of said circuit element inputs and outputs for at least a selected change of condition of a logical combination of said connected circuit element inputs and outputs.

displaying messages describing said traced circuit element inputs and outputs at times when said selected condition is attained,

evaluating the occurrence of at least one selected Boolean relationship among said combination of inputs and outputs,

displaying said trace message whenever the Boolean relationship is satisfied,

said evaluating step comprising the steps of

parsing said Boolean relationship into elemental conditions,

transforming said elemental conditions into event detection circuitry, and

inserting said event detection circuitry into said circuit, the behavior of which is to be simulated.

BEHAVIOR SIMULATOR represented by Block 230. Typically, Block 230 would represent the VHDL simulator hosted on a separate computer or workstation and network coupled to the computer hosting the EVENT MANAGER and SIMULATION CONTROLLER. Upon completion of the behavior-level simulation by Block 230, the dynamic simulation task is de-allocated and the behavior file is parsed for translation to the node-state-time vectors required for the generic simulation response as defined in Table 1 (FIG. 25) by Block 220. This generic simulation response is passed to STATE VECTORS Block 260 for sending to the EVENT MANAGER 100.

Previous Doc Next Doc Go to Doc#

L2: Entry 23 of 28 File: USPT Oct 31, 1989

DOCUMENT-IDENTIFIER: US 4878179 A
**** See image for Certificate of Correction ****
TITLE: Interactive diagnostic methodology and apparatus for microelectronic devices

Detailed Description Text (7):
The SIMULATION CONTROLLER 200 translates the generic simulation directives into simulator-specific input files for loading to predesignated BEHAVIOR SIMULATOR (Block 601), STATE (Block 602), or NODE (Block 603) SIMULATOR. These simulators may be local or remote since they are invoked as dynamic task objects. Following completion of the simulation task, the simulator-specific behavior file is translated into a generic response and returned to the EVENT MANAGER 100 for appropriate stacking. A unique feature of the SIMULATION CONTROLLER 200 is its ability to rapidly create signal flow graph models (see FIG. 6) for switch-level STATE SIMULATOR 602 and node models (see FIG. 5) for NODE CIRCUIT SIMULATOR 603. This is accomplished through a common node-oriented data structure, derived from the specimen's design information, which is specifically tailored for rapid model construction. This data structure preserves topological relationships defining source, sink, and distribution models, including transistor sizes and node-to-node coupling capacitances. These modeling parameters and nodal topologies are derived from an object-oriented symbolic design representation featuring a hierarchy of objects. The lowest level structural objects are clustered to save the topological property of connectedness. The apriori construction of this nodal data structure is performed by the OBJECT CLUSTERING TRANSLATOR 300 prior to diagnostic activities since the specimen's design is considered frozen with regard to behavioral diagnosis.

Detailed Description Text (11):
The use of a generic simulation interface enables a wide range of circuit (node), switch-level (state), and/or behavior simulators to be used in support of diagnostic tactics. Similarly, use of a generic measurement interface allows easy coupling to a variety of beam-probing and stimuli generation apparatus. The generic treatment of simulation and measurement means the bilateral communications between the EVENT MANAGER 100, SIMULATION CONTROLLER 200, and INSTRUMENT CONTROLLER 500 remains constant. The simulation-specific coupling is performed by input/output file format transformation while the apparatus-specific coupling is via apparatus-specific device drivers. This form of generic software-hardware interfacing enables the interactive diagnostic methodology to support a wide spectrum of simulation and beam probing techniques while maintaining a constant interactive diagnostic methodology to the Diagnostician.

Detailed Description Text (71):
Upon receiving a generic simulation directive as defined in Table 1 (FIG. 25) from the EVENT MANAGER 100, INTERPRET DIRECTIVE Block 210 interprets the level of simulation being requested. If a behavior-level simulation is being requested, the control is transferred to BEHAVIOR SIMULATOR-SPECIFIC INTERFACE Block 220. Block 220 parses the generic simulation directive and restructures the model input-output node declarations for suitable retrieving of the behavioral model from the CAD data base. The stimuli pattern is translated into the proper input state sequences for the simulator and the simulation times are set from the given sampling period. Now a dynamic simulation task is properly spawned for execution by the designated

L2: Entry 21 of 28 File: USPT Jan 16, 1996

DOCUMENT-IDENTIFIER: US 5485600 A
TITLE: Computer modelling system and method for specifying the behavior of graphical operator interfaces

Brief Summary Text (65):
The State Table Editor allows the user to specify the behavior of a simulator prototype using a spreadsheet like interface. This spreadsheet is used to specify the desired behavior in terms of behavioral rules. Each behavioral rule is called a "Reaction". A reaction specifies how the prototype should respond to a specified event.

L2: Entry 8 of 28 File: USPT Jul 17, 2001

DOCUMENT-IDENTIFIER: US 6263303 B1
TITLE: Simulator architecture

Brief Summary Text (9):
The typical functional structure of an event-driven simulator is shown in FIG. 2. Event-driven simulators are typically used in simulation of low-level behaviors of the simulated system. Typically, in an event-driven simulator, the behaviors of low level components of a system, which may be individual logic gates captured from a hardware definition language (HDL), or abstractions of logic gates such as used in a register-transfer model, are defined in software modules 16. These software modules are responsive to "events" received by the module from the simulator core 18, and in response generate "events" which are delivered to the simulator core 18 for transfer to other software modules 16. The core 18 references an event-response table 20 to determine where events from a given module are to be transferred, and then transfers the events appropriately. As seen in FIG. 2, a first event(1), causes a module to generate a second event(2), which is transferred to two additional modules.

First Hit Fwd Refs Previous Doc Next Doc Go to Doc#

Generate Collection Print

L2: Entry 2 of 28 File: USPT Nov 30, 2004

DOCUMENT-IDENTIFIER: US 6826518 B1
TITLE: Method for distributed agent-based non-expert simulation of manufacturing process behavior

Brief Summary Text (21):
From the agent perspective, a method for distributed agent-based simulation of manufacturing process behavior, the simulation having a plurality of agents corresponding to individual processes forming a manufacturing technique, comprises the steps of: receiving a message from an agent; identifying in the received message a discrete event selected from the group consisting of a clock tick event, a resources received event, and a request for output production event; causing an associated process to perform an activity in response to the identified event; and, messaging an adjacent agent in response to the identified event.

CLAIMS:

7. A method for distributed agent-based simulation of manufacturing process behavior, the simulation having a plurality of agents corresponding to individual processes forming a manufacturing technique, the method comprising the steps of: receiving a message from an agent associated with one of a plurality of different manufacturing techniques comprising a pull, a push, or a takt manufacturing technique, wherein the agent is responsive to a discrete event selected from the group consisting of a clock tick message, a resources received message, and a request for output production message; identifying in said received message a discrete event selected from the group consisting of a clock tick event, a resources received event, and a request for output production event; causing an associated process to perform an activity in response to said identified event; and, messaging an adjacent agent in response to said identified event.

Previous Doc Next Doc Go to Doc#

Refine Search

Search Results -

Terms	Documents
L1 same event	28

Database:

US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search: L2

Refine Search

Recall Text

Clear

Interrupt

Search History

DATE: Tuesday, December 06, 2005 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L2	L1 same event	28	L2
L1	behavior adj1 simulat$	364	L1

END OF SEARCH HISTORY